AI-driven automation for industrial digitalization: a scalable framework for network discovery and digital twin deployment

AI-driven automation for industrial digitalization: a scalable framework for network discovery and digital twin deployment

1st Alessio Viticchié
AlphaWaves S.r.l.
Torino, Italy
a.viticchie@awaves.it

2nd Alberto Salvatore Colletto
AlphaWaves S.r.l.
Torino, Italy
a.colletto@awaves.it

3th Paolo Bonelli Bassano
AlphaWaves S.r.l.
Torino, Italy
p.bassano@awaves.it

4th Roberto Puntorieri
AlphaWaves S.r.l.
Torino, Italy
r.puntorieri@awaves.it

5th Alessandro Aliberti
Politecnico di Torino
Torino, Italy
alessandro.aliberti@polito.it

***Abstract*—The growing complexity of Industrial Control Systems (ICS) and Operational Technology (OT) networks presents significant challenges in network discovery, device classification, and causal process inference. Traditional methodologies, which depend on manual configurations and static rule-based approaches, often prove inadequate in dynamic industrial environments due to their limited scalability and adaptability.**

This paper introduces an AI-driven agentic framework designed to automate these critical processes. The proposed system employs autonomous AI agents for real-time network scanning, device identification through communication pattern analysis, and inference of process dependencies. By integrating active and passive data collection into the agents' workflow, where they receive insights from these analyses as input, our approach extracts system dynamics without requiring prior domain knowledge of industrial processes.

This methodology advances industrial automation by enabling adaptive, self-optimizing operations, thereby reducing manual intervention and enhancing system visibility. Moreover, it represents a significant step toward the realization of Digital Twins, while also facilitating predictive maintenance and cybersecurity monitoring. Ultimately, this framework offers a scalable and intelligent solution to support the digital transformation of industrial ecosystems.

***Index Terms*—Industrial Digitalization, Agentic AI, Network Discovery, Device Classification, Causal Inference, Industrial Control Systems.**

I. INTRODUCTION

Integrating modern AI-driven methodologies in OT environments is no longer just an improvement but a fundamental requirement driven by the escalating complexity, expanding scale, and critical role of industrial control systems [1]. Conventional OT network management predominantly depends on static configurations, manual device classification, and predefined rule-based approaches. However, these traditional methods are increasingly inadequate for managing the dynamic and evolving nature of industrial infrastructures [2].

Industrial environments are characterized by a diverse mix of devices, the coexistence of both legacy and modern systems, and stringent demands for real-time responsiveness. The rigidity of traditional approaches limits their ability to adapt to rapidly changing conditions, hindering operational efficiency and security [3]. In contrast, AI-based solutions offer the flexibility and intelligence necessary to process vast amounts of data, detect anomalies, and optimize system performance dynamically. By leveraging AI, OT environments can achieve greater adaptability, scalability, and resilience, ensuring efficient and secure management of industrial networks in an era of rapid technological transformation [4].

To address these limitations and to automate the discovery, mapping, and modelling of OT infrastructures, while providing a strong foundation for creating industrial Digital Twins, we implemented an Agentic AI-based framework that identifies devices, reconstructs network topologies, and infers causal relationships between system components. By combining active and passive scanning methods to collect real-time data from industrial networks and by leveraging advanced AI agents, each with specialized roles, the framework provides comprehensive monitoring and adaptive decision-making capabilities reducing the time and expertise required to develop Digital Twins, and enabling faster adoption across diverse industrial contexts.

The remaining sections of this paper are structured as follows: Section II introduces AI-based enabling technology that we have specifically designed to address the limitations of automatic discovery, mapping, and modelling of OT infrastructures. Section III reviews existing methodologies for network discovery and device classification in OT environments, highlighting their limitations and the need for AI-driven approaches. Section IV details the proposed Agentic

Research partially founded by NATIONAL RECOVERY AND RESILIENCE PLAN (PNRR) – MISSION 4 COMPONENT 2 "From Research to Business" INVESTMENT 2.3 "Strengthening and Thematic and Territorial Expansion of Technology Transfer Centers for Industry Segments".

AI framework, describing its architecture, key components, and operational workflow. Section V outlines the experimental setup, including data sources, network configurations, and evaluation metrics. Finally, Section VI summarizes the key findings of this study and discusses potential future research directions for enhancing AI-driven automation in industrial networks.

II. Enabling technologies: AI-driven industrial digitalization

Artificial Intelligence is profoundly transforming various sectors, driving innovation and enabling new automation, optimization, and decision-making possibilities. One of the most significant advancements in this field is the emergence of Large Language Models (LLMs), which have revolutionized how AI systems process information, generate insights and interact with complex environments. This transformation has been largely driven by the introduction of the Transformer architecture, first proposed by Vaswani et al. in [5], which laid the foundation for modern LLMs by enabling efficient parallel processing and enhancing contextual understanding.

LLMs are deep learning models trained on vast datasets, enabling them to understand and generate human-like text, extract meaningful patterns from unstructured data, and perform reasoning tasks with remarkable accuracy. Their adaptability allows them to handle diverse applications, from natural language processing to code generation, knowledge extraction, and decision support.

As depicted in Fig. 1, AI agents leverage LLMs for reasoning, contextual understanding, and decision-making, enabling them to analyze information and determine appropriate actions based on the situation. The study of LLM-powered AI agents has attracted significant research interest in recent years, as documented in [6], [7].

At their core, AI agents consist of three key components:

1) AI Model: Responsible for reasoning and planning, the AI model (often an LLM) interprets inputs, generates responses, and decides on actions. Other models, such as Vision Language Models (VLMs), can also be used to process multimodal inputs.
2) Coordination Layer: Governing the agent's operations, this layer manages identity, objectives, and instructions, ensuring that the agent follows structured workflows. It also handles memory management, allowing the agent to maintain context continuity and improve decision-making over time.
3) Tools: This component extends the agent's capabilities beyond text generation, enabling it to interact with its environment through specialized tools such as API integrations, external functions, and access to knowledge bases, allowing for effective task execution and dynamic adaptability.

The agent's behavior follows a cyclical process in which it continuously ingests information, processes it through internal reasoning, and executes actions using tools based on that reasoning. This loop continues until the agent reaches a predefined goal or a stopping condition.

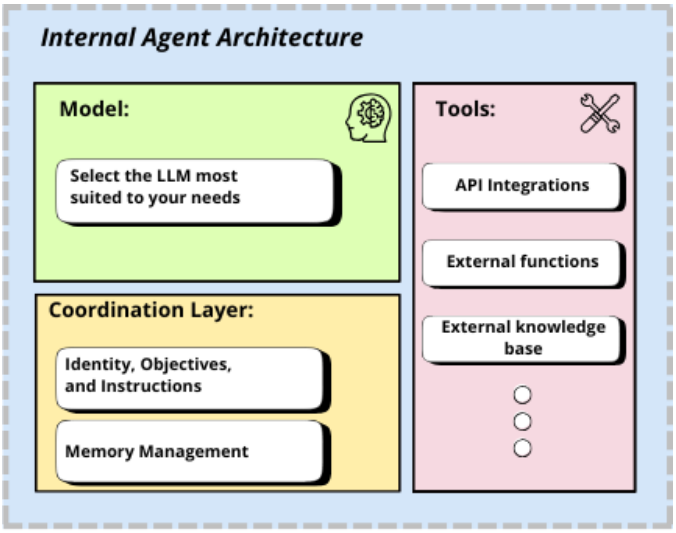

Fig. 1. A visual representation of the internal structure of an AI agent.

AI agents are increasingly being used across various domains, demonstrating their versatility in addressing complex challenges, as exemplified by works such as [8], [9]. Among these applications, they offer a promising solution to the challenges within OT and ICS environments by enabling the coordination of multiple specialized agents to work together efficiently. Through task distribution and structured collaboration, these agents can collectively tackle complex challenges while adapting to dynamic conditions. This approach fosters a more flexible and scalable system, reducing reliance on static, rule-based methodologies.

III. Related Work

Traditional network discovery approaches in industrial control systems and OT environments primarily rely on manual configurations and rule-based tools such as Nmap and PLCScan. While these methods provide a structured approach to device identification and network mapping, they suffer from several limitations, including high setup costs, reliance on static configurations, and a lack of adaptability to dynamic industrial environments [10]. The NIST Guide to ICS Security [11] provides a comprehensive overview of traditional network discovery techniques and their limitations in modern industrial networks. The rigidity of these techniques makes them inefficient in handling the evolving complexity of modern OT networks, where devices and configurations frequently change.

To address these limitations, ML techniques have been explored for device classification, leveraging both supervised and unsupervised learning methods. Recent research, such as [12], has demonstrated how ML models can be trained on network traffic patterns to classify ICS devices. These approaches aim to automate the identification of network components, improving efficiency over manual methods. However, existing

models often struggle with generalization across different OT environments due to the heterogeneity of industrial protocols, variations in device behaviour, and the limited availability of labeled data. The constrained and often proprietary nature of OT networks further complicates the application of generic ML models, resulting in suboptimal classification accuracy in real-world scenarios.

In parallel, causal inference techniques have been applied to cyber-physical systems to analyze dependencies and interactions among components. Methods such as Bayesian Networks and Structural Causal Models offer structured frameworks for modeling cause-and-effect relationships within industrial environments. Additionally, research on passive ICS device discovery and identification through MAC address correlation has explored how passive network analysis can be used to infer network relationships [13].

Despite their potential, these approaches face significant challenges related to scalability, interpretability, and data sparsity. The inherent complexity of industrial processes, coupled with the high dimensionality of network data, makes it difficult to develop accurate causal models without extensive domain knowledge and large volumes of labeled data. Furthermore, traditional causal inference methods often rely on predefined rules or expert-driven assumptions, which can limit their adaptability in highly dynamic settings.

Our work faces these limitations by introducing a system of autonomous AI agents capable of dynamically adapting to new network conditions, thereby overcoming the rigidity of traditional discovery methods. Unlike existing approaches that rely on predefined rule sets or manually curated models, our method harnesses the reasoning, contextual understanding, and adaptability of LLMs to enhance device identification and classification. Additionally, our approach to causal inference does not depend on prior system knowledge, allowing AI agents to infer relationships and dependencies within the network in a more flexible and scalable manner.

The recent rise of LLMs for cybersecurity applications, as documented in [14], further supports the feasibility of AI-driven approaches for network security and anomaly detection. By integrating these capabilities, our work contributes to the development of more adaptive, interpretable, and efficient network management solutions for ICS and OT environments.

IV. Methodology

Our proposed methodology exploits an AI-driven approach that seamlessly integrates both passive and active network analysis, enabling autonomous agents to map, classify, and infer causal relationships within ICS and OT environments. The network data collection strategy follows a structured, multi-phase process designed to ensure comprehensive visibility into communication patterns and network topology within the Company's Local Area Network (LAN). A key objective of this methodology is to passively identify all active hosts within the local network while minimizing any unnecessary network overhead. To achieve this, we deploy DIANA, a proprietary network analyzer which is strategically positioned to monitor traffic across multiple subnets and interfaces. This targeted deployment allows for the efficient extraction of network intelligence while preserving system stability and performance integrity.

Fig. 2 presents the detailed workflow for network traffic extraction using DIANA. The process begins with the integration of DIANA into the target network, ensuring an optimal vantage point for comprehensive data capture. In networks with multiple segregated subnets, additional probes may be deployed, or DIANA may be dynamically repositioned to maintain full visibility across all network segments. Once deployed and operational, DIANA initiates a continuous, cyclic process of monitoring all active connections, communication sessions, and data exchanges—both within the local area network and between the LAN and external networks. The monitored network traffic is first captured and then processed and analyzed using Wireshark, a widely recognized open-source packet analyzer [15], which facilitates the extraction of critical network features. These include communication frequency, message types, protocol usage patterns, and potential anomalies indicative of irregular or suspicious activity.

To maintain a dynamic and continuously updated representation of the network, the analysis is conducted at regular intervals of 10 to 20 seconds. Over time, this iterative process systematically builds a comprehensive inventory of detected hosts and incrementally refines the network topology. Each cycle enriches the DIANA Base Knowledge, enabling it to develop an evolving and increasingly precise understanding of the network structure. Furthermore, when new hosts are detected or existing hosts initiate communication over previously unused ports, an extended in-depth analysis is automatically triggered. This advanced assessment focuses on characterizing the host, identifying active services, and evaluating potential vulnerabilities that could pose security risks. The insights derived from this phase play a critical role in proactive threat detection and risk mitigation. The aggregated data serves as the foundation for advanced processing by AI-driven agents. These models systematically refine, filter, and prioritize network insights, ensuring that only the most relevant and actionable intelligence is retained for further security assessments, anomaly detection, and automated response mechanisms.

This AI-driven methodology, illustrated in Fig. 3, is implemented through a structured multi-agent system designed to sequentially process network traffic data. The data, previously acquired through the DIANA framework and the use of Wireshark, are preprocessed and reorganized into simplified structures to optimize subsequent analytical steps. The multi-agent system is articulated so that each agent performs a specific task, thus allowing the complexity of the problem to be broken down into manageable subtasks and reducing the risk of errors. Each agent processes a partial output that contributes to the final result, allowing several tasks to be executed in parallel. The analysis is performed on batches of connections, the size of which varies depending on the type of object received as input: more complex objects are processed in smaller batch sizes to avoid computational overload of

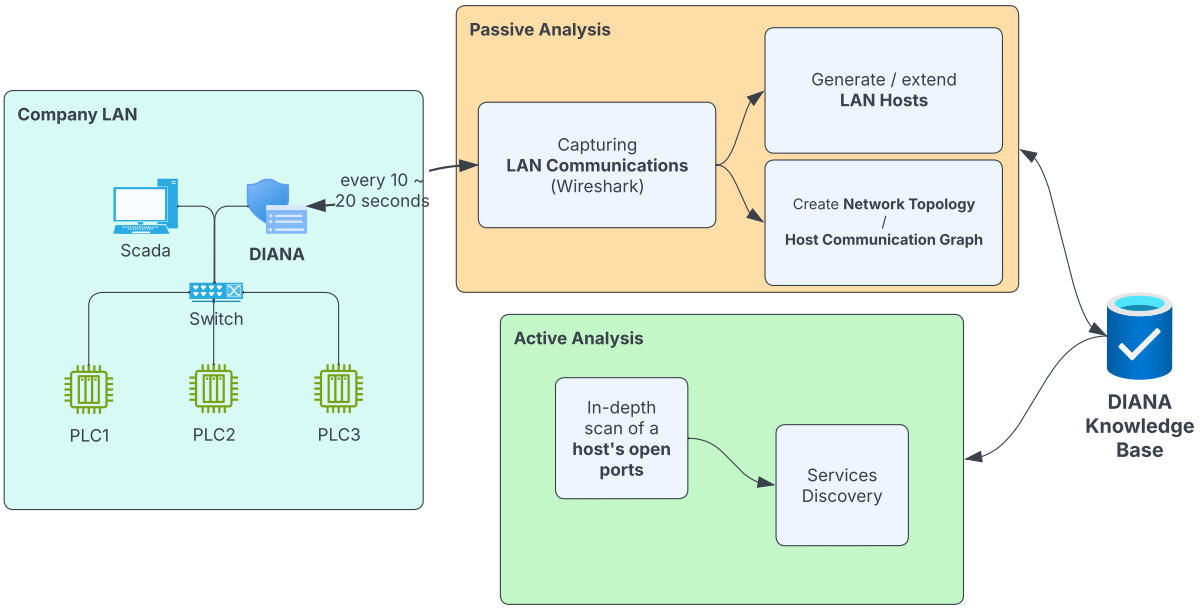

Fig. 2. Gathering Network Traffic with DIANA

the agents. Similarly, each agent receives a limited set of network traffic information, which it analyzes and integrates with previous processing, making the system dynamic and adaptable to changes in network conditions. The workflow consists of two main phases: a first phase of data analysis and processing, implemented through a multi-agent system consisting of three agents with distinct tasks, and a second phase employing an additional multi-agent system, consisting of two agents, aimed at generating the final result.

The specific functions of each agent are described in detail below:

- Agent 1 is responsible for extracting the network topology and identifying hosts, services and connections from network traffic data. The result of this processing is a representation of the topology in JSON format. This step is crucial to exploit the capabilities of the language model in extracting textual information, enabling integration with data previously obtained through DIANA. Thus, the information already possessed about the network topology can be validated, extended or modified, improving the accuracy of the network topology description.
- Agent 2 specializes in analyzing logical flows within the network, tracking packet exchanges to reconstruct dependencies between processes. By examining communications between hosts, the agent deduces the role of each component, identifying key connection criteria, interaction frequency, and the nodes involved. This analysis helps uncover recurring patterns that define the network's operational workflow. For instance, the agent may reveal relationships such as: *"When sensor A detects event X, actuator B is triggered."* Such insights contribute to mapping out a system's behavioral logic. The agent's output includes a comprehensive textual report detailing its hypotheses, inferred flows, and assigned roles for each host, offering a structured overview of network interactions.
- Agent 3 performs traffic-based risk analysis, examining network packets similarly to Agent 2, but with the goal of identifying anomalies, risks, and threats to network security, such as abnormal transmission rates, unauthorized access attempts, or known malware signatures. This phase is a preliminary risk assessment, providing a general overview of the critical issues detected. If potential threats are identified, a more in-depth analysis using dedicated tools is required. The goal of the agent is to support the user in risk prevention by providing an overview of the state of the network. The output generated is a textual report summarizing the vulnerabilities found and explaining their causes.
- Agent 4 is responsible for consolidating the results produced by the previous agents, generating a final output that integrates the various analyses performed. In addition to this synthesis function, the agent can identify any information gaps and formulate targeted questions to fill in the missing information, thus improving the overall quality of the analysis. For example, it might ask, *"I have identified only one subnetwork, are there others?"* This mechanism makes it possible to anticipate the manual integration of additional data in the future. The information provided later will be stored internally by the agent, making it available for future analysis and ensuring the progressive evolution of the network knowledge model.
- Agent 5 is responsible for graphically displaying the results obtained by Agents 1 and 2, transforming them

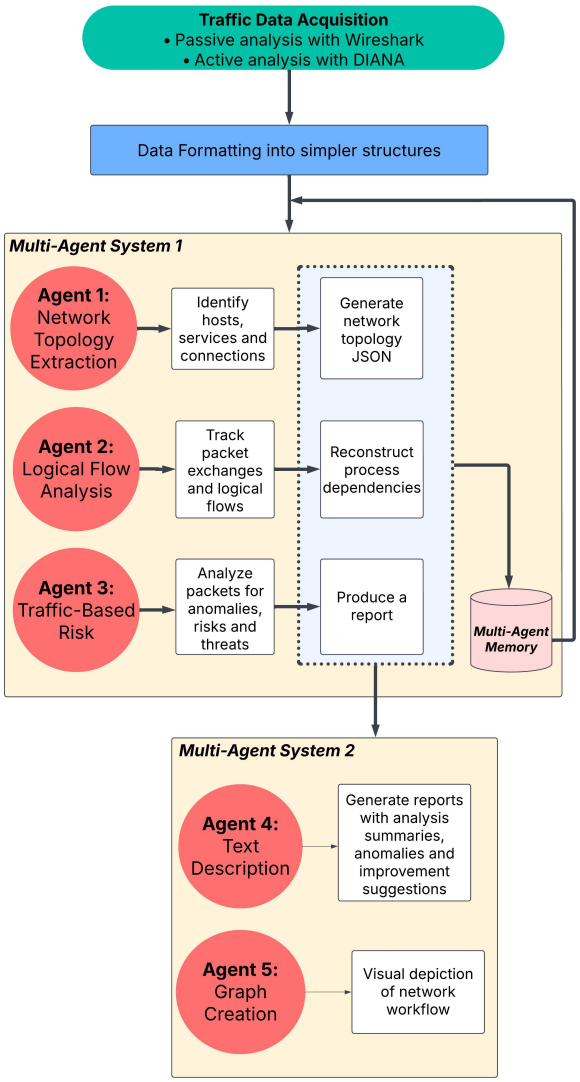

Fig. 3. Multi-agents system workflow

into intuitive representations. Specifically, it exploits the output from Agent 1 to generate a graph in which nodes represent hosts and arcs indicate the connections established based on the detected communications. In parallel, it uses the data provided by Agent 2 to construct a state diagram, which illustrates the hypothesized workflow, highlighting the interactions between hosts and the cause-and-effect relationships between network events. A basic example of this representation is shown in Fig. 4

Agents store the results of their analyses in a shared multi-agent memory, ensuring dynamic and incremental data processing. Because communications are received in separate batches, at varying intervals and with potential differences in the data from previous batches, it is critical that agents can access the results already obtained and integrate them with new information. This approach prevents loss of historical data and reduces redundancy by avoiding duplication of information already captured.

V. Discussion

The implementation of an Agentic AI framework for ICS and OT environments introduces several advantages and challenges. One of the primary strengths of this approach is its self-adaptive and scalable nature, which contrasts sharply with traditional rule-based systems. Unlike static methods that require predefined configurations and frequent manual adjustments, AI agents can dynamically adjust to evolving network conditions, reducing operational overhead and enhancing overall system efficiency. Furthermore, the automation of network discovery and causal inference significantly reduces the manual effort involved in system setup and maintenance, making it a viable solution for large-scale industrial networks.

Despite these advantages, several challenges and limitations must be addressed. Data availability remains a key concern, as AI agents rely on network traffic data to infer relationships between devices. Sparse or incomplete data can negatively impact learning efficiency, leading to potential inaccuracies in classification and inference. Additionally, the real-time constraints of industrial networks pose computational challenges, particularly in large-scale deployments where processing extensive network data can introduce latency. Ensuring the system operates efficiently without disrupting critical industrial processes is a crucial aspect of its practical deployment. Another major challenge is security and robustness, as AI-driven systems are susceptible to adversarial attacks. Malicious actors could manipulate network traffic to mislead AI inferences, potentially compromising network security and stability. Developing defensive mechanisms against adversarial threats is an essential aspect of future research. Lastly, AI agents, relying on LLMs, often exhibit a “black box” nature, making it difficult to fully understand their decision-making processes. This lack of interpretability raises concerns about their accuracy, effectiveness, and potential security risks. Therefore, although AI-driven automation reduces manual effort, it is still prone to errors, requiring robust validation mechanisms to ensure reliability.

In summary, while the Agentic AI framework offers a promising solution for industrial network management, addressing these challenges will be crucial for its successful deployment in real-world ICS and OT environments.

VI. Conclusion

This research underscores the transformative potential of Agentic AI frameworks in industrial environments, providing a foundation for intelligent, automated, and resilient industrial systems. By leveraging AI-driven automation, our approach addresses the limitations of traditional rule-based methods, enabling dynamic network discovery, classification, and causal inference. The integration of passive and active network analysis minimizes the need for predefined models and manual configuration, advancing the broader goal of industrial digitalization.

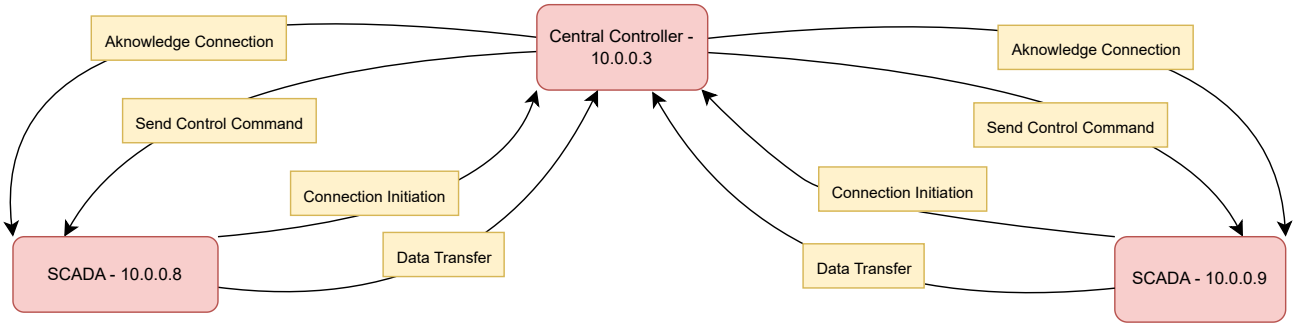

Fig. 4. Example of traffic flow extracted

A significant contribution of this work lies in its potential to enhance Digital Twin development and predictive analytics. AI-driven solutions can support the creation of robust Digital Twins, improving system monitoring, predictive maintenance, and real-time decision-making. Furthermore, the ability of AI agents to infer causal relationships within networks strengthens system diagnostics and process optimization, driving operational efficiency.

Looking ahead, future research could explore the integration of AI-driven threat modeling and vulnerability assessment to further enhance the security and adaptability of ICS and OT infrastructures. Building on the work by Sunder et al. [16], a promising direction would involve extending the Agentic AI framework to model attack graphs and assess vulnerabilities through autonomous agents. This extension could enable dynamic threat modeling, real-time risk assessment, and proactive threat detection, paving the way for more resilient and self-healing industrial systems. By advancing these capabilities, this work contributes to the development of safer, more efficient, and self-sustaining industrial ecosystems.

REFERENCES

[1] D. Mathew, N. Brintha, and J. W. Jappes, "Artificial intelligence powered automation for industry 4.0," in *New horizons for Industry 4.0 in modern business*. Springer, 2023, pp. 1–28.

[2] Y. Maleh, "It/ot convergence and cyber security," *Computer Fraud & Security*, vol. 2021, no. 12, pp. 13–16, 2021.

[3] J. Åkerberg, J. Furunäs Åkesson, J. Gade, M. Vahabi, M. Björkman, M. Lavassani, R. Nandkumar Gore, T. Lindh, and X. Jiang, "Future industrial networks in process automation: Goals, challenges, and future directions," *Applied Sciences*, vol. 11, no. 8, p. 3345, 2021.

[4] R. Yaben, N. Lundsgaard, J. August, and E. Vasilomanolakis, "Towards identifying neglected, obsolete, and abandoned iot and ot devices," in *2024 8th Network Traffic Measurement and Analysis Conference (TMA)*. IEEE, 2024, pp. 1–10.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS)*, 2017.

[6] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin *et al.*, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, p. 186345, 2024.

[7] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou *et al.*, "The rise and potential of large language model based agents: A survey," *Science China Information Sciences*, vol. 68, no. 2, p. 121101, 2025.

[8] Y. Zhang, R. Sun, Y. Chen, T. Pfister, R. Zhang, and S. Arik, "Chain of agents: Large language models collaborating on long-context tasks," *Advances in Neural Information Processing Systems*, vol. 37, pp. 132 208–132 237, 2025.

[9] Z. Gou, Z. Shao, Y. Gong, Y. Shen, Y. Yang, M. Huang, N. Duan, and W. Chen, "Tora: A tool-integrated reasoning agent for mathematical problem solving," *arXiv preprint arXiv:2309.17452*, 2023.

[10] E. Samanis, J. Gardiner, and A. Rashid, "Sok: A taxonomy for contrasting industrial control systems asset discovery tools," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, 2022, pp. 1–12.

[11] K. Stouffer, J. Falco, K. Scarfone *et al.*, "Guide to industrial control systems (ics) security," *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.

[12] I. Chakraborty, B. M. Kelley, and B. Gallagher, "Industrial control system device classification using network traffic features and neural network embeddings," *Array*, vol. 12, p. 100081, 2021.

[13] M. Niedermaier, T. Hanka, S. Plaga, A. von Bodisco, and D. Merli, "Efficient passive ics device discovery and identification by mac address correlation," *arXiv preprint arXiv:1904.04271*, 2019.

[14] H. Xu, S. Wang, N. Li, K. Wang, Y. Zhao, K. Chen, T. Yu, Y. Liu, and H. Wang, "Large language models for cyber security: A systematic literature review," *arXiv preprint arXiv:2405.04760*, 2024.

[15] U. Lamping and E. Warnicke, "Wireshark user's guide," *Interface*, vol. 4, no. 6, p. 1, 2004.

[16] G. Sunder, A. S. Colletto, S. Raimondi, C. Basile, A. Viticchié, A. Aliberti *et al.*, "Enhancing ot threat modelling: An effective rule-based approach for attack graph generation," in *ICSC: Intelligent Cybersecurity Conference*, 2024, p. 2.